

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

KŘIVKY V POČÍTAČOVÉ GRAFICE

BAKALÁŘSKÁ PRÁCE

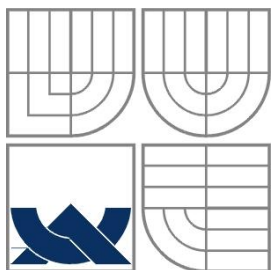
BACHELOR'S THESIS

AUTOR PRÁCE

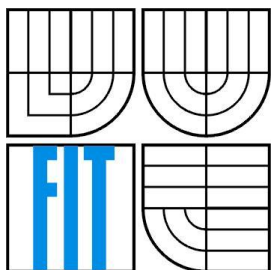
AUTHOR

Luboš Míchal

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

KŘIVKY V POČÍTAČOVÉ GRAFICE

CURVES IN COMPUTER GRAPHICS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

LUBOŠ MÍCHAL

VEDOUCÍ PRÁCE
SUPERVISOR

ING. JIŘÍ VENERA

BRNO 2008

Abstrakt

Tato práce se zabývá návrhem a implementací demonstrační aplikace, která bude sloužit k názorné ukázce křivek používaných v počítačové grafice. Aplikace je vytvořena v jazyce C++ v prostředí Visual Studio 2005. Pomocí této aplikace lze sledovat jednotlivé kroky výpočtu a zadávat řídicí body. Poté je možné pozorovat rozdíly mezi jednotlivými použitými algoritmy. Cílem práce je přiblížit jednotlivé křivky potenciálním zájemcům o toto téma.

Klíčová slova

Křivka, polynom, řídicí polygon, interpolace, aproximace, racionalita, spline, Ferguson, Catmull-Rom, Bézier, Coons, B-Spline, NURBS.

Abstract

This work deal with proposal and implementation demonstrational applicationthat, that will be instrumental towards demonstration curves used in computer graphic. Application is created in language C++ in environment Visual Studio 2005. By the help of those application can be individual steps of calculation watch and set control point. After it is possible watch differences among single used algorithms. Aim work is put near individual curves potential interested person about this theme.

Keywords

Curve, polynomial, control polygon, interpolation, approximation, rationality, spline, Ferguson, Catmull-Rom, Bézier, B-Spline, NURBS.

Citace

Luboš Míchal: Křivky v počítačové grafice. Brno, 2008, bakalářská práce, FIT VUT v Brně.

Křivky v počítačové grafice

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením ing. Jiřího Venery.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Luboš Michal
12. května 2008

Poděkování

Chtěl bych poděkovat ing. Jiřímu Venerovi za jeho odbornou pomoc a vstřícnost při psaní této práce.

© Luboš Michal, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů...

Obsah

Obsah	1
1 Úvod.....	2
2 Křivky	3
2.1 Vlastnosti křivek	3
2.2 Vyjádření křivek.....	4
2.3 Modelování křivek	5
3 Interpolační křivky.....	8
3.1 Fergusonovy kubiky	8
3.2 Catmull-Rom spline	10
4 Aproximační křivky	11
4.1 Bézierovy křivky	11
4.1.1 Bézierovy kubiky	11
4.1.2 Obecné Bézierovy křivky	13
4.1.3 Algoritmus de Casteljau	14
4.2 Racionální Bézierovy křivky	15
4.3 Coonsovy kubiky.....	17
4.4 Spline křivky	18
4.4.1 B-Spline	18
4.5 NURBS	20
5 Návrh aplikace	23
5.1 Návrh z pohledu uživatele	23
5.2 Návrh z pohledu programátora	23
6 Závěr	25
Literatura.....	26

1 Úvod

Použití výpočetní techniky v oblasti modelování, návrhu a optimalizaci je dnes již nezbytnost. O to se také velkou měrou zasloužil prudký nárůst strojírenského průmyslu. Snad největší uplatnění našlo počítačové modelování v automobilovém a leteckém průmyslu. Obrovským podílem se na tomto vývoji podílí i svět filmu, kde se počítačová animace stala nezbytnou součástí. Digitální podoba přináší nesporné výhody. Těmi zásadními pak jsou přímá vizualizace, možnost kontroly a nenákladné provádění testů (nedochází k žádným materiálním škodám), rychlé provádění změn, optimalizace a mnoho dalších.

Při návrhové a designérské práci se pak nejčastěji používá generování dat. To znamená, že model je na počítači přímo vytvořen, přičemž je očekáván vysoký stupeň kvality. Pro dosažení tohoto požadavku musí být počítač schopen detailně vymodelovat požadovaný objekt. Právě křivky jsou jedním ze základních stavebních prvků při modelování obecných objektů a tvarů.

Křivky jsou definovány funkcemi, které lze matematicky vyjádřit. To ale není vhodné pro modelování objektu, neboť odhadnout matematickou rovnici na základě tvaru je velmi obtížné. Mnohem lepší variantou je generování křivky v grafickém prostředí pomocí několika bodů, podle nichž bude počítač danou křivku modelovat.

Modelování křivek se stalo samostatnou vědní disciplínou výpočetní geometrie, přičemž nejvíce se o přínos do tohoto oboru zasloužili právě automobilový a letecký průmysl. Ti jsou stále „hnacím motorem“ pro tuto tematiku, neboť jejich stále se zvyšující požadavky neustále nutí lidi zdokonalovat tuto technologii.

Při modelování křivek je důležité alespoň přibližně vědět, jaké požadavky má výsledná křivka splňovat. Protože metod pro aplikaci křivky je několik a každá křivka má své specifické vlastnosti a chování, je třeba zvážit, kdy je vhodné jakou křivku využít.

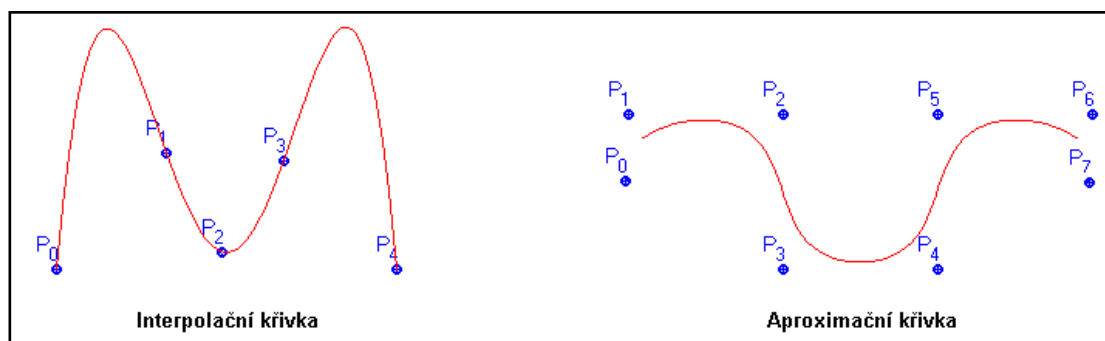
Aplikací, které pak umožňují běžnému člověku styk s modelováním křivek nejrůznějších tvarů je několik. Ať už se jedná o klasické grafické editory, které použil snad každý z nás, nebo o specializované systémy, pomocí nichž vznikají detailní digitální podoby různých objektů (například CAD systémy).

2 Křivky

Křivky jako takové jsou rozděleny dle několika kritérií do různých skupin. V počítačové grafice jsou těmito kritérii především typ křivky, způsob vyjádření křivky a algoritmus, jakým je daná křivka modelována.

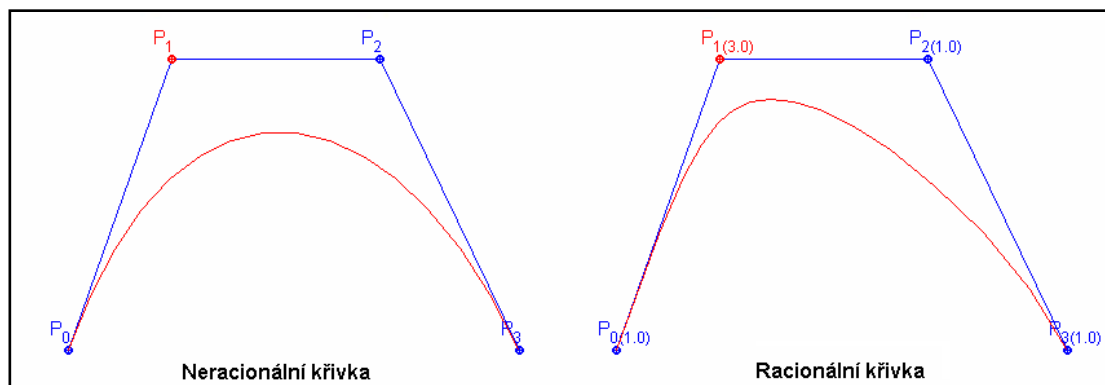
2.1 Vlastnosti křivek

Mezi základní vlastnosti křivek patří rozdělení na interpolační a aproximační křivky. Zatímco křivka interpolační prochází body řídícího polygonu, křivka aproximační je „proložena“ mezi řídícími body.



Obr. 2.1 Interpolační x Aproximační křivka

Další podstatná vlastnost při modelování je racionalita a neracionalita křivky, přičemž racionální křivka je taková, která bere v potaz jako parametr bodu nejen jeho pozici, ale i jeho tzv. váhu. Čím vyšší má bod váhu oproti ostatním, tím více se k němu vymodelovaná křivka přiklání. Neracionální, jak již název vypovídá, nebere v potaz váhu bodu. Tedy všechny body mají stejný váhový koeficient (obvykle hodnota jedna) [lit ref 2]. Jako příkladné znázornění je na obr. 2.2 zobrazena křivka, jejíž druhý bod P_1 má váhový koeficient roven 3 oproti ostatním.



Obr. 2.2 Neracionální x Racionální křivka

2.2 Vyjádření křivek

V oblasti počítačové grafiky jsou křivky obvykle vyjádřeny nějakou rovnicí, která je posléze generativně zobrazována. Toto vyjádření může být obecně trojího druhu – implicitní, explicitní nebo nejčastěji používané parametrické vyjádření [lit ref 2].

Křivka zadaná explicitně je zadaná jako funkce ve tvaru:

$$y = f(x) \quad (2.1)$$

Toto vyjádření je pro modelování křivek v počítačové grafice nevhodné, neboť lze pracovat pouze s křivkami, které jsou zároveň funkcemi. Tudíž musí platit, že pro jednu hodnotu x z definičního oboru existuje pouze jedna odpovídající funkční hodnota $f(x)$.

Křivka zadaná implicitně:

$$F(x,y) = 0 \quad (2.2)$$

Protože toto vyjádření neumožňuje postupný výpočet křivky, opět není vhodné pro potřeby počítačové grafiky. Je ovšem vhodné pro jiné aplikace, jako například nalezení průsečíku křivky s jinou křivkou nebo práce s oblastí vymezenou touto křivkou.

Parametrické vyjádření křivky:

$$\begin{aligned} x &= x(t), \\ y &= y(t) \end{aligned} \quad (2.3)$$

Vektorové parametrické vyjádření:

$$P(t) = [x(t), y(t)] \quad (2.4)$$

Přičemž parametr t se většinou volí z intervalu $< 0, 1 >$. Pro $t = 0$ nebo $t = 1$ jsou pak vektorovým zápisem vyjádřeny krajní body. Výhodou tohoto vyjádření je závislost pouze na jediném parametru, nevýhodou pak nerovnoměrné rozložení bodů na křivce.

Jednou z dalších výhod parametrického vyjádření křivky je snadný výpočet jejích tečných vektorů. Ty jsou důležité při navazování jednotlivých křivek (segmentů) za účelem jedné složené křivky. Při navazování segmentů sledujeme především spojitost v bodu navázání (takový bod, ve kterém jeden segment končí a další začíná), přičemž rozlišujeme dva druhy spojitosti, parametrickou a geometrickou. Značí se C^n respektive G^n , kde n určuje stupeň spojitosti.

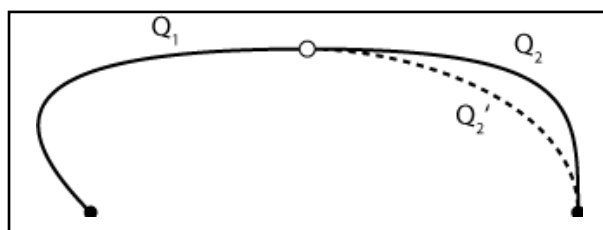
Říkáme, že křivka je C^n spojitá, pokud má spojitě derivace podle parametru t až do n -tého řádu. Spojitost C^0 zajistíme ztotožněním koncového bodu jednoho segmentu s počátečním bodem segmentu následujícího. Spojitost C^1 je zajištěna ztotožněním tečných vektorů obou segmentů v bodu jejich navázání. Všechny další stupně spojitosti je možné analogicky zajistit totožností odpovídajících stupňů derivací v bodu navázání [lit ref 2]. Z fyzikálního hlediska znamená spojitost C^0 spojitou dráhu bodu, směr jeho pohybu, rychlost i zrychlení se však mohou skokově měnit.

Spojitost stupně C^1 zaručí, že směr pohybu a velikost rychlosti se skokově nemění, ale pro velikosti zrychlení to však nelze zaručit. To, že se ani velikost zrychlení nebude měnit skokově, zaručuje spojitost C^2 .

Kromě parametrické spojitosti se používá také již zmiňovaná geometrická spojitost. Podmínky pro spojitost G^0 jsou stejné jako pro spojitost C^0 , přičemž taková spojitost je zajištěna, platí-li, že tečný vektor, kde $t = 1$ je roven k -násobku tečného vektoru, kde $t = 0$.

$$\mathbf{q}'_1(1) = k * \mathbf{q}'_2(0) \quad (2.5)$$

Na rozdíl od spojitosti C^1 může bod skokově měnit rychlost, směr jeho pohybu se však skokově nemění. Vyšší geometrickou spojitost než je spojitost G^1 většinou nepoužíváme. Segmenty Q_1 a Q_2 na následujícím obrázku jsou C^1 spojitě, segmenty Q_1 a Q_2' jsou „jen“ G^1 spojitě.



Obr. 2.3 Vizualní rozdíl mezi C^1 a G^1 spojitostí

2.3 Modelování křivek

V počítačové grafice jsou nejčastější variantou křivky polynomiální ($P_n(t) = a_0 + a_1 + \dots + a_n t^n$). Z takovýchto křivek jsou po částech skládány křivky polynomiální [lit ref 3]. Protože složitější tvary křivek s sebou nesou i použití polynomů vysokých stupňů, používá se místo jednoho polynomu více kratších segmentů nižšího stupně. Při použití těchto kratších segmentů se pak sleduje spojitost v místech navázání.

Pro segmenty se nejčastěji používá polynomů třetího stupně – kubik [lit ref 2]. Ty dokáží zachytit širokou škálu tvarů a jejich výpočet není příliš náročný. Při použití takovýchto segmentů lze zaručit spojitost C^2 , která je často požadována při modelování v CAD systémech. Křivky vyššího stupně by pak mohli způsobit nechtěné vlnění a jsou mnohem náročnější na výpočet.

Jednotlivé složky parametricky zadané kubiky pro křivku pak budou mít tvar:

$$\begin{aligned} x(t) &= a_x t^3 + b_x t^2 + c_x t + d_x, \\ y(t) &= a_y t^3 + b_y t^2 + c_y t + d_y, \\ z(t) &= a_z t^3 + b_z t^2 + c_z t + d_z, \end{aligned} \quad (2.6)$$

Zapsáno v maticovém tvaru:

$$Q(t) = TC = [t^3, t^2, t, 1] \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix} \quad (2.7)$$

V tomto zápisu je v matici C dvanáct parametrů určujících výsledný tvar segmentu. Z takového zápisu ale nejsme schopni odhadnout výsledný tvar křivky. Pro pohodlnější tvarování by bylo vhodné ovlivňovat ty parametry, které se dají přímo zobrazovat, tedy řídicí body, popř. tečné vektory.

Za tímto účelem je vhodné rozdělit charakteristiky daných tříd křivek na dvě skupiny, kde jedna skupina bude představovat ty vlastnosti, které jsou společné celé skupině křivek dané třídy, druhá skupina pak bude představovat vlastnosti, které jsou pro každou křivku individuální.

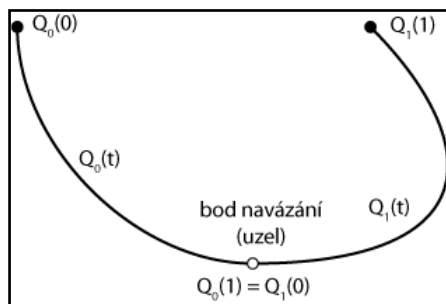
V tomto případě můžeme psát $C = MG$, kde M je matice konstant o rozměru 4×4 nazývaná báзовou maticí a čtyřprvkový vektor G se nazývá vektor geometrických podmínek. Polynomiální báze daná součinem TM představuje skupinu polynomů, která je společná pro všechny křivky dané třídy. Vektor geometrických podmínek pak obsahuje ty parametry, které si samy můžeme ovlivňovat za účelem dosažení požadovaného tvaru křivky, tedy řídicí body nebo tečné vektory.

Kubika pak bude vypadat takto:

$$Q(t) = TMG = [t^3, t^2, t, 1] \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix} \quad (2.8)$$

Každý z polynomů polynomiální báze pak určuje, která z geometrických podmínek G_i bude mít větší vliv na tvar výsledné křivky na jejím začátku, která ve vnitřní části a která na jejím konci.

Samotné modelování segmentů křivky, potažmo křivky celé pak probíhá tak, že je zadáno několik bodů řídicího polygonu a na jejich základě je pomocí zvoleného matematického aparátu stanovena výsledná podoba křivky. Další možností pak je zadávat přímo tečné vektory na koncích jednotlivých segmentů složené křivky. Při modelování je obvykle možno klást podmínky na hladkost navázání, kvalitu vykreslené křivky atd.



Obr. 2.4 Křivka složená ze dvou segmentů

U modelované křivky pak obecně sledujeme požadované vlastnosti [lit ref 6]:

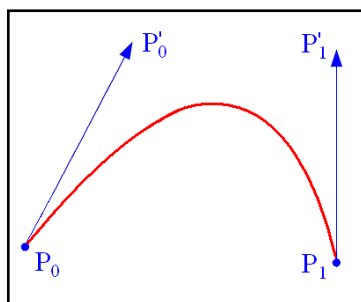
- ***invariance k lineárním transformacím*** – tedy že pokud provedeme rotaci řídicích bodů, nebude to mít vliv na tvar křivky.
- ***konvexnost obálky*** – sledujeme, zda vygenerovaná křivka leží v konvexní obálce svých řídicích bodů, a to buď celá křivka, nebo jenom její jednotlivé segmenty.
- ***lokalita změn*** – zajímá nás, zda při změně polohy jednoho řídicího bodu dojde k změně tvaru celé křivky, nebo jen její části (okolí změněného bodu).
- ***interpolace krajních bodů*** – zajímá nás, zda křivka prochází či neprochází počátečním a koncovým bodem řídicího polygonu.

O tom, jaké požadované vlastnosti modelovaná křivka má a o jejích dalších vlastnostech rozhoduje především algoritmus použitý pro vykreslení křivky.

3 Interpolační křivky

3.1 Fergusonovy kubiky

Fergusonova kubika občas také nazývána jako kubika Hermitovská je nejčastěji používanou interpolační křivkou. Tato metoda, kterou používal J. C. Ferguson již roku 1964 při konstruování křivek v leteckém průmyslu, je řízena dvěma koncovými body P_0 a P_1 , určujícími polohu dané kubiky a dvěma tečnými vektory P'_0 a P'_1 [lit ref 1].



Obr. 3.1 Fergusonova kubika

Koncové body jsou pro tuto křivku významné, neboť jimi modelovaná křivka prochází, vektory pak slouží pro určení „vyklenutí“ křivky. Čím je velikost tohoto vektoru větší, tím více se k němu modelovaná křivka přiklání.

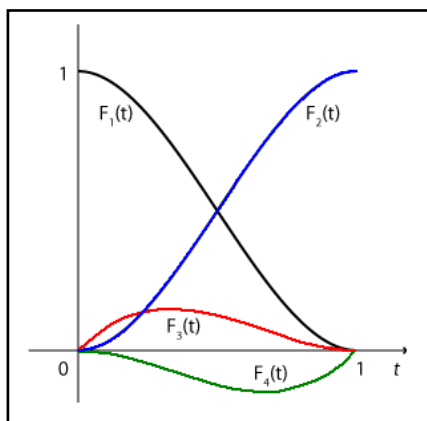
Fergusonova křivka je zadána body P_0 a P_1 , a dvěma tečnými vektory P'_0 a P'_1 ze kterých je odvozena výsledná rovnice pro výpočet kubiky ve tvaru:

$$P(t) = P_0F_1(t) + P_1F_2(t) + P'_0F_3(t) + P'_1F_4(t), \quad (3.1)$$

kde F_1 , F_2 , F_3 a F_4 jsou kubické Hermitovské polynomy, pro které platí:

$$\begin{aligned} F_1(t) &= 2t^3 - 3t^2 + 1, \\ F_2(t) &= -2t^3 + 3t^2, \\ F_3(t) &= t^3 - 2t^2 + t, \\ F_4(t) &= t^3 - t^2 \end{aligned} \quad (3.2)$$

přičemž $t \in [0, 1]$. Položíme-li $t = 0$, bude $P(0) = P_0$, pokud pak položíme $t = 1$, vyjde nám analogicky $P(1) = P_1$. Křivka tedy prochází krajními body, což potvrzuje vlastnost interpolace krajních bodů. Pokud derivujeme $P(t)$ podle t a opět dosadíme $t = 0$ a $t = 1$, zjistíme, že $P'(0) = P'_0$ a $P'(1) = P'_1$, tedy že tečné vektory v krajních bodech jsou identické s těmi, co zadal uživatel.



Obr. 3.2 Kubické Hermitovské polynomy

Pak maticový zápis fergusonovi kubiky vypadá:

$$Q(t) = TMG = [t^3, t^2, t, 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_0' \\ P_1' \end{bmatrix} \quad (3.3)$$

Z uvedených rovnic vyplývá, že $P(t)$ je kubikou, neboť se jedná o součet polynomů třetího stupně. Položíme-li $P'_0 = P'_1 = P_1 - P_0$, bude touto metodou aproximována úsečka. Spojitost prvních derivací (hladkost křivky) pak bude zaručena identitou vektorů.

Výhodou této metody je snadné navazování sousedních segmentů (koncový bod prvního segmentu je shodný s počátečním bodem segmentu následujícího) a hladkost vymodelované křivky, pokud je zaručeno ztotožnění tečných vektorů ztotožněných bodů (identita vektorů P'_1 prvního segmentu a P'_0 segmentu následujícího). Nevýhodami pak neinteraktivní a neintuitivní řízení tvaru křivky a nelokální změna tvaru při změně pozice jednoho bodu. Nevýhodou je rovněž poměrně obtížná editace tečných vektorů při práci v trojrozměrném prostoru [lit ref 2].

Obecně se doporučuje tuto metodu použít jen v případech, kdy jsou známy tečné vektory a to jak jejich směr, tak i jejich velikost v jednotlivých uzlech (body, ve kterých se napojují jednotlivé segmenty křivky) složené modelované křivky.

3.2 Catmull-Rom spline

Tuto metodu navrhli E. Catmull a R. Rom. Ti v jedné ze svých publikací dokázali, že při matematickém vyjádření B-Spline křivky lze nahradit kontrolní body parametrickou funkcí t , přičemž tato partikulární funkce má stejné interpolační vlastnosti jako řídící body. Třída B-spline křivek nazývaná Catmull-Rom spline patří mezi interpolační křivky, přičemž díky svým vlastnostem našli hlavní použití při definování dráhy objektů v počítačové animaci [lit ref 1].

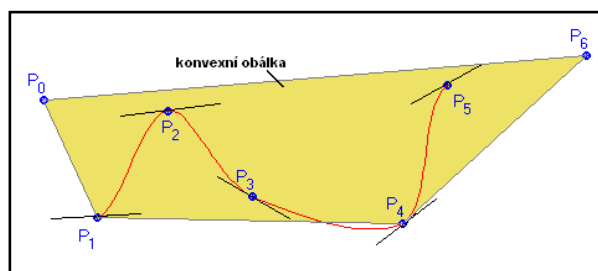
Tato křivka je definována posloupností bodů P_0, P_1, \dots, P_n . Vychází z bodu P_1 a končí v bodě P_{n-1} , tedy takováto křivka nezaručuje sama o sobě vlastnost interpolace krajních bodů.

Křivka se vypočítá ze vztahu:

$$Q(t) = \frac{1}{2} [t^3, t^2, t, 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_{i-3} \\ P_{i-2} \\ P_{i-1} \\ P_i \end{bmatrix} \quad (3.4)$$

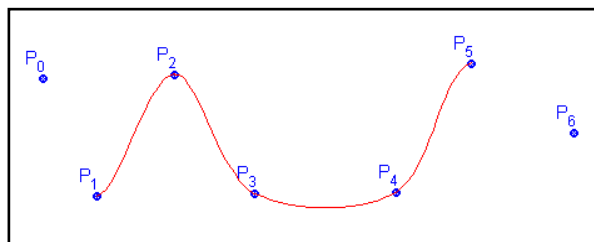
Přičemž výpočet probíhá postupně, tedy nejprve s body P_0, P_1, P_2, P_3 , poté s body P_1, P_2, P_3, P_4 a tak dále až do výpočtu s body $P_{n-3}, P_{n-2}, P_{n-1}, P_n$.

Mezi vlastnosti takového spline patří, že tečný vektor v bodě P_n je rovnoběžný s vektorem $P_{i-1} - P_{i+1}$. To určuje přirozený tvar křivky. Pro zaručení interpolace krajních bodů se používá zdvojení počátečního bodu P_0 a koncového bodu P_n . Nevýhodou této metody pak je, že výsledný spline obecně neleží v konvexní obálce.



Obr. 3.3 Spline s vyznačenými tečnami v řídících bodech a zobrazenou konvexní obálkou

Výsledný Catmull-Rom spline může vypadat např. takto:



Obr. 3.4 Příklad Catmull-Rom spline

4 Aproximační křivky

4.1 Bézierovy křivky

Teoretický základ těmto aproximačním křivkám položil v letech 1959 – 1962 P. E. Bézier, když vyvíjel programový nástroj UNISURF pro návrh křivek a ploch u francouzské firmy Renault. Nezávisle na něm používal také tuto metodu pro návrh aproximace křivek také P. de Casteljau. Roku 1970 pak R. Forrest ukázal na souvislost mezi Bézierovým aparátem a teorií Bernsteinových polynomů. Bézierovy křivky jsou pravděpodobně nejpopulárnějšími křivkami používanými mimo jiné i pro návrh počítačových fontů (písmen).

4.1.1 Bézierovy kubiky

Bézierovy kubiky jsou nejčastější variantou Bézierových křivek používaných v počítačové grafice. Kubika, tedy třetí stupeň polynomu je totiž nejnižším stupněm, který může zajistit spojitost až do třídy C^2 , což je pro použití v počítačové grafice naprosto dostačující [lit ref 2].

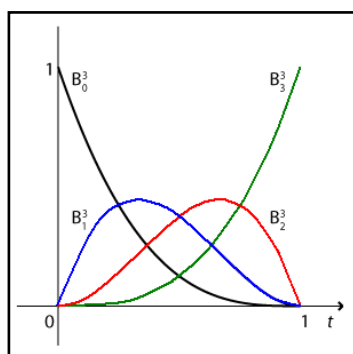
Bézierova kubika je zadána čtyřmi body P_0 , P_1 , P_2 a P_3 , přičemž modelovaná křivka vychází z bodu P_0 a končí v bodě P_3 . Body P_1 a P_2 určují vyklenutí.

Bézierova kubika je definována vztahem:

$$P(t) = P_0 B_0(t) + P_1 B_1(t) + P_2 B_2(t) + P_3 B_3(t) = \sum_{i=0}^3 P_i B_i(t), \quad (4.1)$$

kde $t \in < 0, 1 >$ a B_0 , B_1 , B_2 a B_3 jsou kubické polynomy tvaru:

$$\begin{aligned} B_0(t) &= (1 - t)^3, \\ B_1(t) &= 3t(1 - t)^2, \\ B_2(t) &= 3t^2(1 - t), \\ B_3(t) &= t^3 \end{aligned} \quad (4.2)$$



Obr. 4.1 Kubické Bernsteinovy polynomy

Bézierovu kubiku lze také zapsat maticově:

$$Q(t) = T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \quad (4.3)$$

Položíme-li $t = 0$ a dosadíme do rovnice pro výpočet Bézierovy kubiky, vyjde nám, že $P(0) = P_0$, analogicky pak pro $t = 1$ bude $P(1) = P_3$, tedy tato kubika prochází krajními body. Pokud derivujeme $P(t)$, dostaneme:

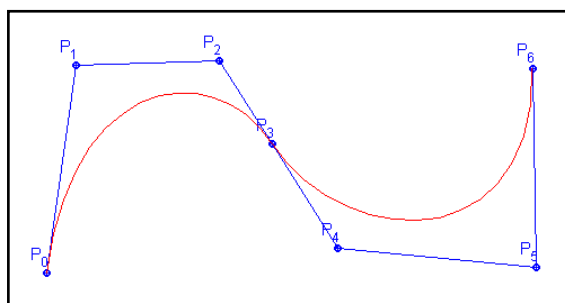
$$P'(t) = \sum_{i=0}^3 P_i B'_i(t) \quad (4.4)$$

Pokud znovu dosadíme $t = 0$ a $t = 1$, zjistíme, že:

$$\begin{aligned} P'(0) &= 3(P_1 - P_0), \\ P'(1) &= 3(P_2 - P_3). \end{aligned} \quad (4.5)$$

Tedy že tečné vektory mají vždy směr spojnice dvojice krajních bodů, přičemž jejich velikost bude rovna trojnásobku vzdálenosti bodů.

Hladké navázání dvou sousedních segmentů křivky bude zajištěno, pokud bude zaručena jejich spojitost (poslední bod předchozího oblouku je identický s prvním bodem segmentu následujícího, nebo křivka prochází prvním a posledním bodem) a pokud budou identické tečné vektory [lit ref 1]. Z toho lze odvodit, že druhý bod následujícího oblouku je určen posledními dvěma body oblouku předchozího.



Obr. 4.2 Příklad Bézierovy kubiky

4.1.2 Obecné Bézierovy křivky

Zobecněním Bézierových kubik jsou obecné Bézierovy křivky. Tato křivka řádu n vznikne z $n + 1$ bodů řídicího polygonu P_0, P_1, \dots, P_n . Takováto křivka je definovaná vztahem:

$$P(t) = P_0 B_0^n(t) + P_1 B_1^n(t) + P_2 B_2^n(t) + \dots + P_n B_n^n(t) = \sum_{i=0}^n P_i B_i^n(t), \quad (4.6)$$

kde B_i^n jsou Bernsteinovy polynomy n -tého stupně ve tvaru:

$$B_i^n = \binom{n}{i} t^i (1-t)^{n-i}, \quad (4.7)$$

přičemž $t \in \langle 0, 1 \rangle$ a $i = 0, 1, \dots, n$.

Pokud do vztahu definujícího obecnou Bézierovu křivku dosadíme $t = 0$, bude $P(0) = P_0$, analogicky pak při dosazení $t = 1$ vyjde $P(1) = P_n$. Z toho lze soudit, že i obecná Bézierova křivka prochází krajními body řídicího polygonu (interpolace krajních bodů).

Po derivování $P(t)$ získáme vztah:

$$P'(t) = \sum_{i=0}^n P_i B_i'^n(t) \quad (4.8)$$

Pokud opět provedeme dosazení za t hodnotami 0 a 1, vyjde nám:

$$\begin{aligned} P'(0) &= n(P_0 - P_1), \\ P'(1) &= n(P_{n-1} - P_n). \end{aligned} \quad (4.9)$$

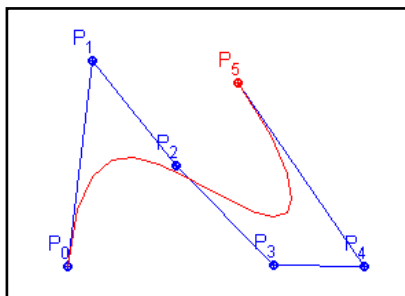
Tedy že tečné vektory mají směr shodný se spojnicí vždy dvojice krajních bodů a jejich velikost je rovna n -násobku jejich velikosti, kde n je stupeň aproximační křivky [lit ref 1].

Pro zajištění dostatečné hladkosti křivky (navazování segmentů) platí stejná pravidla jako pro kubické Bézierovy křivky (je-li identický poslední bod předchozího segmentu s počátečním bodem segmentu navazujícího, je složený segment spojitý, navíc je-li zároveň tento bod středem úsečky spojující předposlední bod prvního segmentu s druhým bodem segmentu následujícího, je spojení hladké).

Použité Bernsteinovy polynomy B_i^n mají následující vlastnosti:

- Pro všechna $i, n \in \mathbb{N} \cup \{0\}$ a $t \in \langle 0, 1 \rangle$ je $B_i^n(t) \geq 0$ - jsou nezáporné
- $\sum_{i=0}^n B_i^n(t) = 1$ pro $t \in \langle 0, 1 \rangle$ - křivka leží v konvexní obálce
- $B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t)$ - motivace algoritmu de Casteljau

Z toho, že křivka leží v konvexní obálce bodů řídícího polygonu, plyne podstatná metoda nasazení této metody v technické praxi, neboť nežádané zvlnění křivek je vždy v mezích konvexní obálky. Třetí vlastnost pak byla motivací pro P. de Casteljau pro tvorbu rekursivního algoritmu (viz. Níže).



Obr. 4.3 Příklad obecné Bézierovy křivky

Bézierovy křivky jsou často díky své snadné tvorbě a snadné manipulovatelnosti používány v CAD systémech. Pro změnu tvaru křivky totiž stačí pouhá změna polohy bodu. V počítačových aplikacích je navíc výpočet takovýchto křivek velmi rychlý, neboť se používají pouze operace sčítání a násobení (výjimku tvoří kombinační čísla v Bernsteinových polynomech, která jsou spočtena dopředu) [lit ref 4].

4.1.3 Algoritmus de Casteljau

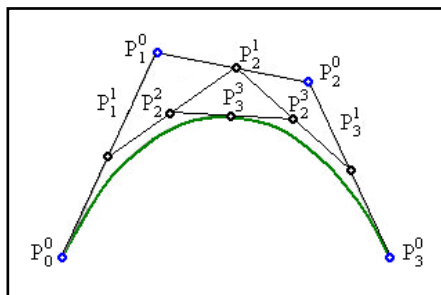
Tento rekursivní algoritmus vytvořený Paulem de Casteljau, je ve své podstatě geometrickým postupem, jak vygenerovat bod na Bézierově křivce pro danou hodnotu t , pokud známe body řídícího polygonu. Jde tedy o jiný přístup, který využívá stejné vstupní podmínky a přitom dává stejný výsledek jako při použití analytické metody využívající Bernsteinových polynomů (viz. Kapitola 4.1.1).

Algoritmus de Casteljau vypadá následovně:

$$P_i^k = (1 - t)P_{i-1}^{k-1} + tP_i^{k-1} , \quad (5.0)$$

kde $k = 1, \dots, n$ a $i = 0, \dots, n-k$, přičemž n značí počet řídících bodů.

Pokud si výpočet bodu na Bézierově křivce znázorníme, zjistíme, že se ve skutečnosti nejedná o nic jiného nežli o postupné dělení úseček řídícího polygonu na dvě části v poměru t a $1 - t$, kde $t \in <0, 1>$ (viz. Obr. 4.4). Počet nově vytvořených bodů se v každém kroku zmenšuje o jedna až do té chvíle, kdy nám zůstane pouze jediný bod a to bod hledaný.



Obr. 4.4 Znáznornění výpočtu de Casteljau

Výpočet probíhá následovně:

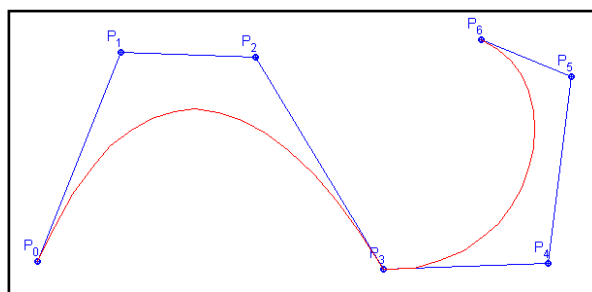
$$P_0^0$$

$$P_1^0 \quad P_1^1$$

$$P_2^0 \quad P_2^1 \quad P_2^2$$

$$P_3^0 \quad P_3^1 \quad P_3^2 \quad P_3^3 = P(t)$$

Tento algoritmus nejvíce oceníme při vykreslování Bézierovy křivky na diskretním rastru. Při výpočtech vznikají nové segmenty podle zvoleného kroku, otázkou ovšem zůstává, jak ideálně zvolit velikost kroku aby vykreslená křivka byla dostatečně hladká a zároveň výpočet nebyl zbytečně zdouhový. Další věcí, která snižuje kvalitu tohoto algoritmu, je rozložení bodů. Tento algoritmus totiž rovnoměrně rozprostírá body, což znamená, že pracuje stejně s rovnou částí křivky i se zvlněnou. Tuto vlastnost lze v metodě zefektivnit tak, že výpočet se bude provádět tak dlouho, dokud vypočtený úsek křivky nebude dostatečně rovná [lit ref 1].



Obr. 4.5 Bézierova křivka metodou de Casteljau

4.2 Racionální Bézierovy křivky

Tato poměrně nová metoda používaná v grafických aplikacích je založena, jak již název napovídá, na zobecnění Bézierových křivek. Při klasickém zadání uživatel zadá řídicí polygon a program následně podle jeho požadavku buď proloží body křivkou patřičného řádu, nebo výslednou křivku složí z křivek řádu nižšího. Přičemž pro změnu tvaru křivky musel uživatel změnit polohu bodu, což nemusí být vždy výhodou. Tento problém se pak zejména projevuje při práci v trojrozměrném prostoru, protože identifikace takového bodu není jednoduchá. Proto se používá metoda, která každému bodu řídicího polygonu přiřadí reálné číslo, jehož změnou se mění tvar křivky.

Racionální Bézierova křivka je určena posloupností bodů P_0, P_1, \dots, P_n a posloupností reálných čísel $\omega_0, \omega_1, \dots, \omega_n$, kde ω_i je váha i -tého bodu řídicího polygonu. Bézierova racionální křivka stupně n je pak určena vztahem:

$$P(t) = \frac{\sum_{i=0}^n \omega_i P_i B_i^n(t)}{\sum_{i=0}^n \omega_i B_i^n(t)} = \sum_{i=0}^n P_i R_i^n, \quad (5.1)$$

kde R_i^n jsou racionální Bernsteinovy polynomy. Ty lze vyjádřit:

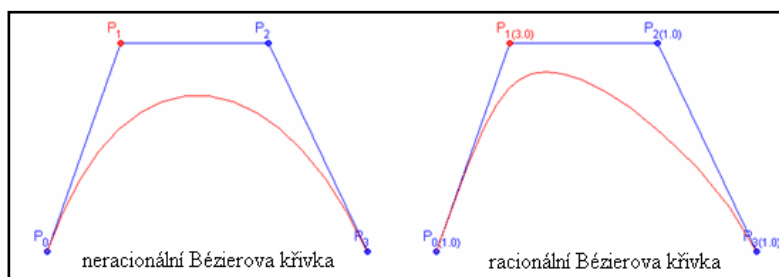
$$R_i^n = \frac{\omega_i B_i^n(t)}{\sum_{j=0}^n \omega_j B_j^n(t)}. \quad (5.2)$$

Pro tyto polynomy platí obdobná pravidla jako pro klasické Bernsteinovy polynomy, tedy že:

- Pro všechna $i, n \in \mathbb{N} \cup \{0\}$ a $t \in \langle 0, 1 \rangle$ je $R_i^n(t) \geq 0$ - jsou nezáporné
- $\sum_{i=0}^n R_i^n(t) = 1$ pro $t \in \langle 0, 1 \rangle$ - křivka leží v konvexní obálce
- Je-li $\omega_i = 1, i = 0, 1, \dots, n$, pak $R_i^n = B_i^n$

Z posledního vztahu vyplývá, že racionální Bézierovy křivky jsou skutečně zobecněním Bézierových křivek. Tedy lze dokázat, že pokud položíme $t = 0$ a dosadíme do rovnice pro výpočet racionální Bézierovy křivky, vyjde nám, že $P(0) = P_0$ a analogicky pak pokud dosadíme $t = 1$ získáme $P(1) = P_1$ [lit ref 1]. Z tohoto lze opětovně vyvodit, že i racionální Bézierova křivka splňuje podmínky interpolace krajních bodů.

Na následujícím obrázku je uveden rozdíl mezi klasickou Bézierovou křivkou a racionální Bézierovou křivkou, kde bod P_0 má zvýšený váhový koeficient.



Obr. 4.6 Neracionální x Racionální Bézierova křivka

Největším přínosem této metody je, jak již bylo zmíněno možnost měnit tvar křivky bez nutnosti změny polohy bodů řídicího polygonu. Tato metoda je také velmi vhodná pro modelování křivek, které obsahují prudké zlomy.

4.3 Coonsovy kubiky

Další uvedenou metodou pro modelování křivek jsou Coonsovy kubiky. Tato kubika má některé odlišné vlastnosti oproti doposud uvedeným křivkám. Metodu definoval S. A. Coons a její uplatnění je především díky dobrým geometrickým vlastnostem hlavně v navrhování ploch.

Coonsova kubika je definována stejně jako kubika Bézierova čtyřmi řídicími body P_0, P_1, P_2, P_3 , a kubickými polynomy C_0, C_1, C_2 a C_3 . Výsledná kubika se pak dá spočítat ze vztahu [lit ref 3]:

$$P(t) = \frac{1}{6} [P_0 C_0(t) + P_1 C_1(t) + P_2 C_2(t) + P_3 C_3(t)], \quad (5.3)$$

přičemž:

$$\begin{aligned} C_0(t) &= (1-t)^3, \\ C_1(t) &= 3t^3 - 6t^2 + 4, \\ C_2(t) &= -3t^3 + 3t^2 + 3t + 1, \\ C_3(t) &= t^3, \end{aligned} \quad (5.4)$$

kde $t \in [0, 1]$.

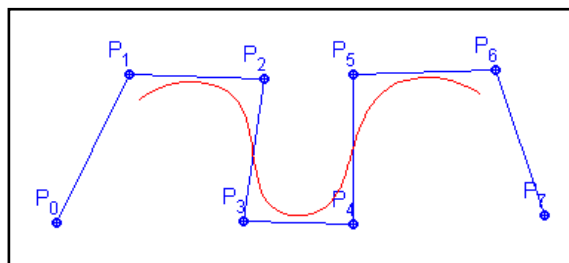
Maticový zápis této rovnice:

$$Q(t) = \frac{1}{6} T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \quad (5.5)$$

Pokud dosadíme za $t = 0$, vyjde nám, $P(0) = \frac{P_0 + 4P_1 + P_2}{6}$ a pokud dosazení za $t = 1$ bude $P(1) = \frac{P_1 + 4P_2 + P_3}{6}$. Bod $P(0)$ leží v první třetině těžnice jdoucí z bodu P_1 do středu strany P_0P_2 . Tento bod je nazýván antitěžištěm, v tomto případě trojúhelníku $P_0P_1P_2$. To samé platí i pro bod $P(1)$ a trojúhelník $P_1P_2P_3$ [lit ref 2].

Pro tuto metodu platí, že:

- tečna v $P(0)$ je rovnoběžná se stranou P_0P_2 a její velikost je rovna polovině strany P_0P_2 .
- je-li bod P_1 středem úsečky P_0P_2 , vychází aproximační křivka z bodu P_1 .
- je-li $P_0 = P_1$, říkáme, že se jedná o dvojnásobný bod a aproximační křivka vychází z bodu: $\frac{P_0P_2}{6} = \frac{P_1P_2}{6}$.
- je-li $P_0 = P_1 = P_2$, říkáme, že se jedná o trojnásobný bod a křivka vychází z něho.



Obr. 4.7 Coonsův kubický B-spline

Největší výhoda této metody se projeví až při použití skládání aproximačních křivek. Pokud budeme uvažovat řídicí polygon P_0, P_1, \dots, P_n a křivku budeme skládat pomocí Coonsových oblouků tak, že jeden oblouk použijeme body P_0, P_1, P_2, P_3 , pro další pak body P_1, P_2, P_3, P_4 atd., získáme křivku, která se nazývá B-Spline [lit ref 3].

Vlastností B-Spline křivky je, že má ve všech vnitřních bodech spojitost druhého řádu. Z hlediska modelování takovéto křivky je výhodné i to, že dochází pouze k lokální změně tvaru křivky při změně pozice některého z bodů řídicího polygonu.

4.4 Spline křivky

Spline křivky jsou jistým zobecněním polynomiálních interpolací a aproximací. Tyto křivky jsou složeny z jednotlivých segmentů. Tyto složené křivky z jednotlivých segmentů jsou nazývány anglickým slovem spline, protože jsou matematickým modelem chování pružných křivítek (anglicky spline), která se dříve používala k výrobě trupů lodí. V praxi mají zásadní význam kubické spline křivky, takže všechny doposud uvedené kubiky se k jejich konstrukci dají použít, přičemž nejčastěji používanou je kubika Coonsova a tedy Coonsův kubický B-Spline.

4.4.1 B-Spline křivky

B-Spline křivky poprvé zavádí N. Lobanovský již v 19. století. Roku 1946 použil I. J. Schoenberg B-Spline křivek pro vyhlazování statistických dat a dává tím základy moderní teorii spline aproximací. M. Cox a C. de Boor nezávisle na sobě objevili roku 1972 rekurentní vztah pro výpočet B-Spline. Tato teorie byla poté použita pro výpočet parametrických B-Spline křivek.

Obecná B-Spline křivka stupně k je určena rovnicí:

$$Q(t) = \sum_{i=0}^n P_i N_i^k(u), \quad (5.6)$$

kde $N_i^k(u)$ jsou normalizované b-spline báze funkce (někdy také nazývané de Boor funkce).

Pro tyto funkce pak platí:

$$N_i^k(u) = \frac{u-u_i}{u_{i+k}-u_i} N_i^{k-1}(u) + \frac{u_{i+k+1}-u}{u_{i+k+1}-u_{i+1}} N_{i+1}^{k-1}(u), \quad (5.7)$$

kde:

$$N_i^0(u) = 1, \text{ pro } u \in \langle u_i, u_{i+1} \rangle,$$

$$N_i^0(u) = 0, \text{ pro } u \notin \langle u_i, u_{i+1} \rangle.$$

Hodnoty parametrů u_i se obvykle nazývají uzly. Poté se mluví o tzv. uzlovém vektoru parametrů:

$$U = (u_0, u_1, \dots, u_k, u_{k+1}, \dots, u_{m-k}, \dots, u_m), \quad (5.8)$$

kde $u_i < u_{i+1}$, $m = n+k+1$.

Uzlový vektor se nazývá periodickým (uniformním), jestliže platí, že:

$$u_{i+1} - u_i = konst.$$

B-Spline křivky mají následující vlastnosti [lit ref 6] [lit ref 2]:

- $N_i^k(u) \geq 0$ - polynomy mají nezápornou hodnotu
- $\sum_{i=0}^n N_i^k(u) = 1$, pro $u \in \langle u_0, u_m \rangle$ - platí jednotkový součet *
- Při dělení nulou je výsledek roven nule
- Křivka prochází počátečním a koncovým bodem řídicího polygonu a dotýká se počáteční a koncové hrany řídicího polygonu

* Jednotkový součet potvrzuje, že křivka leží v konvexní obálce určené body P_0, P_1, \dots, P_n .

Výsledný stupeň B-Spline křivky lze zvolit oproti Bézierově křivce, kde je pevně určen. Je-li zvolena uniformní neperiodická parametrizace a pro $k = 0, \dots, n$ platí [lit ref 3]:

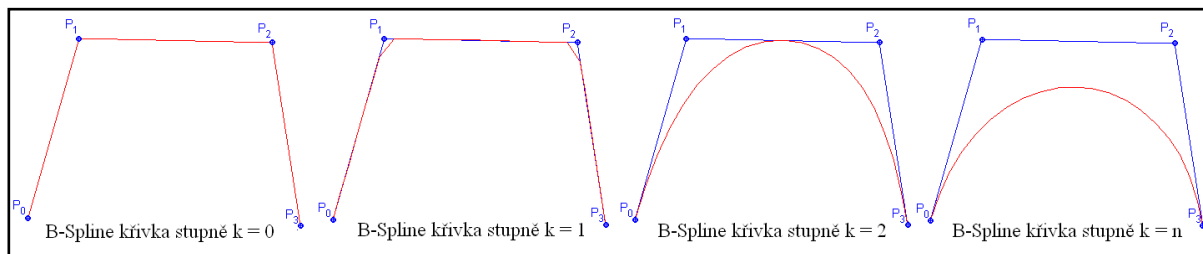
pro $k = 0$ dostáváme pouze body řídicího polygonu,

pro $k = 1$ dostáváme právě řídicí polygon,

pro zvyšující stupeň se křivka vzdaluje od řídicího polygonu,

pro $k = n$ získáme stejnou křivku jako při použití Bézierovy křivky pro daný polygon.

Lokální změna tvaru je zajištěna tím, že báze funkce $N_i^k(u)$ nabývají nenulových hodnot jen na intervalu $\langle u_i, u_{i+k+1} \rangle$. To znamená, že změníme-li polohu jednoho bodu řídicího polygonu, pak se změní pouze ta část křivky, pro kterou $u \in \langle u_i, u_{i+k+1} \rangle$. Tato změna ovlivní $k+1$ segmentů křivky. Čím je tedy stupeň B-Spline křivky vyšší, tím větší část křivky se změní při změně jednoho bodu řídicího polygonu, přičemž pro $k = n$ dojde ke změně celé křivky.



Obr. 4.8 Zobrazení B-spline křivky při postupném zvyšování stupně křivky

4.5 NURBS

Neuniformní racionální B-Spline křivky nazývané NURBS (z anglického *Non Uniform Rational B-Spline*) jsou jedny z nejpoužívanějších typů parametrických křivek. NURBS křivky a plochy se stali populárními zejména díky své podpoře pro matematicky přesné modelování kuželoseček (kružnice, elipsa, ale i část kružnice). Tyto křivky jsou dvojným zobecněním B-Spline křivek, přičemž termín neuniformní (neperiodický) je odvozen od vzdálenosti uzlů ve smyslu parametru t , která nemusí být u těchto křivek konstantní, tedy $t_{i+1} - t_i \neq konst$. Racionalita, jak již bylo dříve vysvětleno, naznačuje, že tento algoritmus pracuje pouze s polohou bodu, nýbrž bere v potaz i jeho váhový koeficient.

Křivka NURBS je určena $n + 1$ body řídicího polygonu, uzlovým vektorem U délky $m + 1$ a stupněm křivky k . Uzlový vektor procházející prvním a posledním bodem řídicího polygonu má obvykle tvar:

$$U = \{0, 0, \dots, 0, t_{p+1}, \dots, t_{m-p-1}, z, z, \dots, z\}.$$

Jednotlivé uzly musí splňovat podmínku $t_i \leq t_{i+1}$, tedy že jejich hodnota je neklesající a zároveň může platit, že $t_i - t_{i+1} \neq konst$. Sekvence nul na začátku, stejně tak jako sekvence hodnot z na konci vektoru (obvykle voleno $z = 1$) má násobnost $k + 1$ [lit ref 1]. Mezi počtem uzlů $m + 1$, stupněm křivky k a počtem bodů řídicího polygonu $n+1$ platí vztah:

$$m = n + k + 1. \quad (5.9)$$

Rovnice křivky NURBS má tvar:

$$Q(t) = \sum_{i=0}^n P_i R_{i,k}(t), \quad (6.0)$$

kde $R_{i,k}(t)$ je racionální B-Spline báze, kterou zjistíme:

$$R_{i,k}(t) = \frac{\omega_i N_{i,k}(t)}{\sum_{j=0}^n \omega_j N_{j,k}(t)}. \quad (6.1)$$

Rovnici lze rovněž zapsat ve tvaru:

$$Q(t) = \frac{\sum_{i=0}^n \omega_i P_i N_{i,k}(t)}{\sum_{i=0}^n \omega_i N_{i,k}(t)}, \quad (6.2)$$

kde ω_i je váha i -tého bodu řídicího polygonu a $N_{i,p}(t)$ jsou normalizované B-Spline bazové funkce určené vztahem:

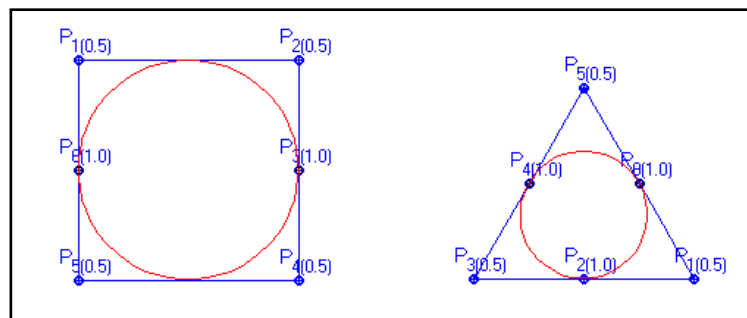
$$N_{i,0}(t) = \begin{cases} 1 & \text{pro } t_i \leq t \leq t_{i+1} \\ 0 & \text{jinde} \end{cases}$$

$$N_{i,k}(t) = \frac{(t - t_i)}{(t_{i+1} - t_i)} N_{i,k-1}(t) + \frac{(t_{i+k+1} - t)}{(t_{i+k+1} - t_{i+1})} N_{i,k+1}(t)$$

Křivky NURBS mají následující vlastnosti [lit ref 7] [lit ref 1]:

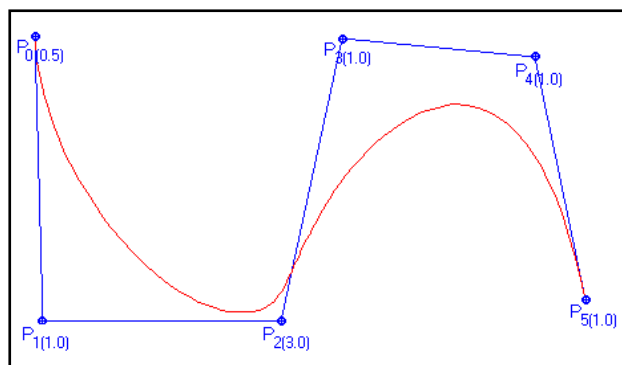
- pokud uzlový vektor bude zadán tak, jak zde bylo definováno, bude NURBS křivka procházet počátečním i koncovým bodem řídicího polygonu.
- křivka NURBS leží v konvexní obálce svého řídicího polygonu, stejně tak její segmenty leží v konvexních obálcích svých řídicích polygonů. Změna polohy resp. váhy některého z bodů řídicího polygonu tedy má vliv pouze na část křivky.
- jsou invariantní vůči transformacím a to především vůči rovnoběžnému a středovému promítání.
- umožňují přesné vyjádření kuželoseček.
- pro uzlový vektor $U = \{0, 0, \dots, 0, 1, 1, \dots, 1\}$, kde počet nul a jedniček je stejný, jsou normalizované B-Spline báze rovny Bernsteinovým polynomům a NURBS je pak racionální Bézierovou křivkou. Pokud hodnot ω všech bodů řídicího polygonu je rovna 1, je NURBS křivka neracionální B-Spline. Pokud je navíc stupeň křivky roven n , je zároveň křivkou Bézierovou.

Zejména jako důležitá se jeví vlastnost umožnit přesné vyjádření kuželoseček. Ta je podstatná pro tvorbu jednotlivých datových struktur reprezentujících jak klasické geometrické struktury, jakými jsou například kružnice nebo parabola, tak i volné tvary.



Obr. 4.9 Vyjádření kružnice pomocí NURBS

Po vykreslení může NURBS křivka vypadat například i takto:



Obr. 5.0 NURBS křivka stupně 3 se zvýšenou vahou bodu P_2

Výhodou NURBS křivek je jejich snadná editace uživatelem. Ten může polohu bodů řídícího polygonu, jejich váhu, stupeň křivky a případně i uzlový vektor při použití neinteraktivního modelování. Protože v interaktivním modelování stačí k dosažení jakéhokoli tvaru pouze změna polohy bodu řídícího polygonu nebo změna jeho váhy.

5 Návrh aplikace

5.1 Návrh z pohledu uživatele

Protože aplikace má být především demonstrační, je navržena tak, aby běžný uživatel neměl žádné problémy při zadávání vstupních parametrů a výběru algoritmu, který křivku vykreslí. Uživateli je zároveň umožněno měnit tvar vykreslené křivky ať už změnou polohy bodu řídicího polygonu, přidáním nebo odebráním bodu, nebo v některých případech měnit váhu některého z bodů řídicího polygonu (racionální křivky) a volit stupeň křivky (B-Spline a NURBS křivky). Výpočet, pomocí kterého je křivka vykreslena, je zároveň možno jednoduchým způsobem krokovat a to tak, že samotný uživatel graficky vidí, jak se křivka graficky mění z úsečky na křivku. Pro srovnání různých způsobů výpočtu křivky lze pro zadaný polygon řídicích bodů lze měnit algoritmus pro výpočet podle toho, jaké metody chce uživatel porovnávat.

5.2 Návrh z pohledu programátora

Tato aplikace byla vytvořena ve vývojovém prostředí Visual Studia 2005 jazykem C++, přičemž byly použity standardní knihovny a knihovna GLUT. Tato knihovna tvoří doplněk ke knihovně OpenGL, která umožňuje lepší práci s okny a událostmi nad nimi.

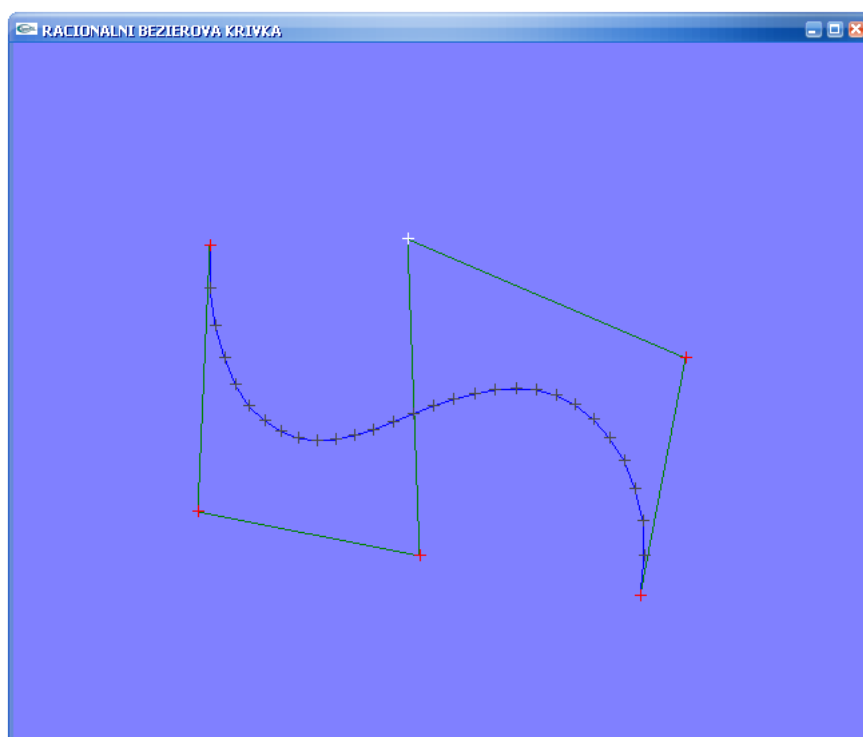
Protože aplikace provádí především výpočty jednotlivých bodů vybranými metodami, používá hlavně matematické operace pro řešení rovnic, které jsou jednotlivě implementovány do příslušných funkcí, stojí za zmínku jen několik věcí. Jedna z nich je vytvořená struktura pro bod obecně. Tato struktura bodu obsahuje základní věci, jakými jsou x-ová a y-ová poloha bodu a váha bodu. Dále pak obsahuje proměnou *activity*, která sděluje, zda je daný bod aktivní, přičemž aktivní znamená, že tento bod byl pomocí tlačítka myši vybrán pro další práci s ním (smazání, zvýšení/snížení váhy nebo změna polohy). Tuto vlastnost pak uživatel rozpozná tak, že bod je zobrazen barvou odlišnou ostatním. Uživateli se zobrazují i body vypočtené křivky, ale ty jsou pouze informativní a plní funkci demonstrační, tudíž není možné je ani přesouvat, ani mazat.

V projektu se používají zejména dva globální vektory bodů – do prvního se ukládají body řídicího polygonu, ve druhém jsou pak ukládány vypočítané body výsledné křivky. Další vektor je použit pouze při metodách pracujících s uzlovým vektorem, tedy u metod B-Spline a NURBS. Vykreslení pak probíhá tak, že tyto vypočítané body jsou spojeny pomocí tzv. polyčáry, což není nic jiného než vykreslení úsečky vždy mezi dvěma sousedními body.

Křivka pak sama o sobě má dva parametry, které jsou v projektu definovány jako globální proměnné – parametr, který udává kvalitu vykreslené křivky (počet proložených bodů) a parametr udávající stupeň křivky (tento argument se uplatňuje pouze u některých algoritmů). Tyto parametry může uživatel nastavovat sám pomocí předdefinovaných kláves.

Projekt obsahuje jak funkce pro samotný výpočet bodů křivky dle stanoveného algoritmu, pak funkce zajišťující vše okolo. To je například reakce na stisk klávesy, stisk tlačítka myši, pohyb myši se stisknutým tlačítkem, změnu velikosti okna atd.

Při tvorbě této demonstrační aplikace byl kladen ohled především na uživatele. Tedy aby pro něj bylo pokud možno co nejjednodušším a nejsrozumitelnějším způsobem demonstrováno, jak která křivka pracuje a jaké jsou odlišnosti mezi implementovanými algoritmy. Vlastnosti křivek si pak může uživatel sám ověřit a to jak pomocí změn některých hodnot bodu řídicího polygonu, tak také změnou vlastností křivky.



Obr. 5.1 Vytvořená demonstrační aplikace

6 Závěr

Cílem tohoto projektu bylo přiblížit vývoj a současný trend používaných algoritmů pro výpočet bodů křivky na základě zadaných atributů. Aplikace, která vznikla jako součást projektu, uživateli poskytuje možnost shlédnout, jak jednotlivé algoritmy pracují a jak křivka jimi vygenerovaná vypadá a jak se liší od ostatních algoritmů.

Přínosem tohoto projektu je, že i když se uživatel nevyzná v tom, jak pracují počítače nebo v matematických funkcích jednotlivých metod, je schopen pomocí nastavování různých atributů křivek a bodů řídicího polygonu rozpoznat, jak daná metoda provádí modelování křivky krok za krokem. Dále pak má možnost sledovat rozdíly, které vznikají, pokud pro tytéž hodnoty bodů řídicího polygonu zvolí jiný algoritmus pro výpočet bodů křivky.

Protože tento projekt se zabývá demonstrací křivek v rovině, určitě by bylo zajímavé rozšířit toto téma o křivky v 3D prostoru, neboť právě trojrozměrné objekty nás obklopují. Vzhledem k tomu, že křivky v počítačové grafice jsou i úzce spojovány s plochami, nabízí se jako jedna z dalších možností pokračování projektu zaměření se na popis a demonstraci ploch používaných při modelování v počítačové grafice. Toto téma je v dnešní době velmi aktuální a hojně využívané, neboť pro modelování objektů nebo různých scén jsou křivky a plochy základními prvky.

Počítačová grafika jako taková nachází v dnešní době obrovské uplatnění v mnoha odvětvích a její možnosti využití jsou obrovské. A protože nároky na zpracování obrazu jsou čím dál větší, musí se i metody pro jeho zpracování zdokonalovat.

Literatura

- [1] Alexandr Lubomír, Výuka počítačové grafiky cestou www [Diplomová práce], Brno.
- [2] <http://herakles.zcu.cz/education/zpg/cviceni.php?no=8>
- [3] <http://cs.wikibooks.org/wiki/Geometrie>
- [4] http://www.cazone.net/dp/granty/grant_3/teorie/main.htm
- [5] <http://atrey.karlin.mff.cuni.cz/projekty/vrr/doc/man/progman/Rational-Bezier-curves.html>
- [6] <https://www.fit.vutbr.cz/study/courses/IZG>
- [7] <http://www.root.cz/clanky/format-x3d-a-zahadna-zkratka-nurbs>

Seznam příloh

Příloha 1. Manuál aplikace

Příloha 2. CD s aplikací, zdrojovými texty a technickou zprávou